NYCU Pattern Recognition, Homework 3

Deadline: May 4, 23:59

Part. 1, Coding (80%):

In this coding assignment, you need to implement the Decision Tree, AdaBoost and Random Forest algorithm by using only NumPy, then train your implemented model by the provided dataset and test the performance with testing data. Find the sample code and data on the GitHub page
https://github.com/NCTU-VRDL/CS_AT0828/tree/main/HW3

Please note that only NumPy can be used to implement your model, you will get no points by simply calling sklearn.tree.DecsionTreeClassifier.

1. (5%) Gini Index or Entropy is often used for measuring the "best" splitting of the data. Please compute the Entropy and Gini Index of this array
   `np.array([1,2,1,1,1,1,2,2,1,1,2]).`

   ```
   Gini of data is  0.4628099173553719
   ```

   ```
   Entropy of data is  0.9456603046006401
   ```

2. (10%) Implement the Decision Tree algorithm (CART, Classification and Regression Trees) and train the model by the given arguments, and print the accuracy score on the test data. You should implement **two arguments** for the Decision Tree algorithm, 1) **Criterion**: The function to measure the quality of a split. Your model should support "gini" for the Gini impurity and "entropy" for the information gain.
   2) **Max_depth**: The maximum depth of the tree. If Max_depth=None, then nodes are expanded until all leaves are pure. Max_depth=1 equals split data once

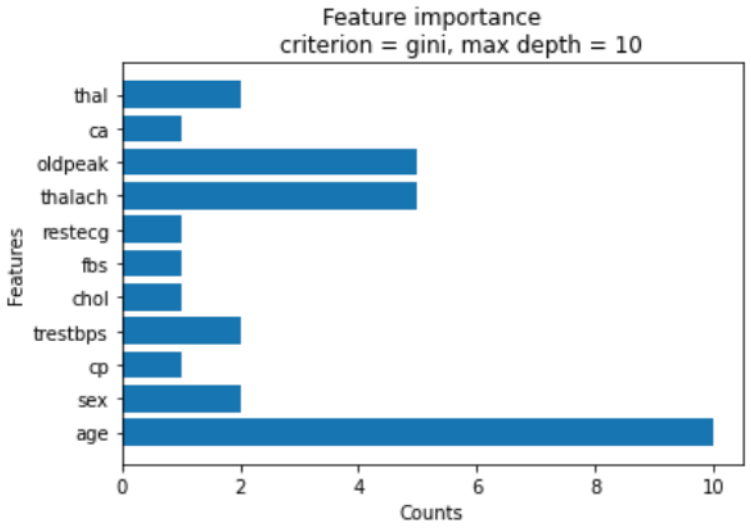
   2.1. Using Criterion='gini', showing the accuracy score of test data by Max_depth=3 and Max_depth=10, respectively.

   ```
   criterion: gini, max_depth: 3, Accuracy = 0.83
   criterion: gini, max_depth: 10, Accuracy = 0.78
   ```

   2.2. Using Max_depth=3, showing the accuracy score of test data by Criterion='gini' and Criterion='entropy', respectively.

   ```
   criterion: gini, max_depth: 3, Accuracy = 0.83
   criterion: entropy, max_depth: 3, Accuracy = 0.8
   ```

3. (5%) Plot the feature importance of your Decision Tree model. You can use the model from Question 2.1, max_depth=10. (You can use simply counting to get the feature importance instead of the formula in the reference, more details on the sample code. **Matplotlib** is allowed to be used)



4. (15%) Implement the AdaBoost algorithm by using the CART you just implemented from question 2. You should implement **one argument** for the AdaBoost.
   1) **N_estimators**: The number of trees in the forest.

   4.1. Showing the accuracy score of test data by n_estimators=10 and n_estimators=100, respectively.

```
n_estimators: 10, Accuracy = 0.8
```

```
n_estimators: 100, Accuracy = 0.8
```

5. (15%) Implement the Random Forest algorithm by using the CART you just implemented from question 2. You should implement **three arguments** for the Random Forest.
   1) **N_estimators**: The number of trees in the forest.
   2) **Max_features**: The number of features to consider when looking for the best split
   3) **Bootstrap**: Whether bootstrap samples are used when building trees

   5.1. Using Criterion='gini', Max_depth=None, Max_features=sqrt(n_features), Bootstrap=True, showing the accuracy score of test data by n_estimators=10 and n_estimators=100, respectively.

```
criterion: gini, estimators = 10, bootstrap = True.
Accuracy of test-set = 0.79
```

```
criterion: gini, estimators = 10, bootstrap = False.
Accuracy of test-set = 0.78
```

```
criterion: gini, estimators = 100, bootstrap = True.
Accuracy of test-set = 0.8
```

5.2. Using Criterion='gini', Max_depth=None, N_estimators=10, Bootstrap=True, showing the accuracy score of test data by Max_features=sqrt(n_features) and Max_features=n_features, respectively.

```
criterion: gini, estimators = 10 with random features.
Accuracy of test-set = 0.79
```

```
criterion: gini, estimators = 10 with all features.
Accuracy of test-set = 0.79
```

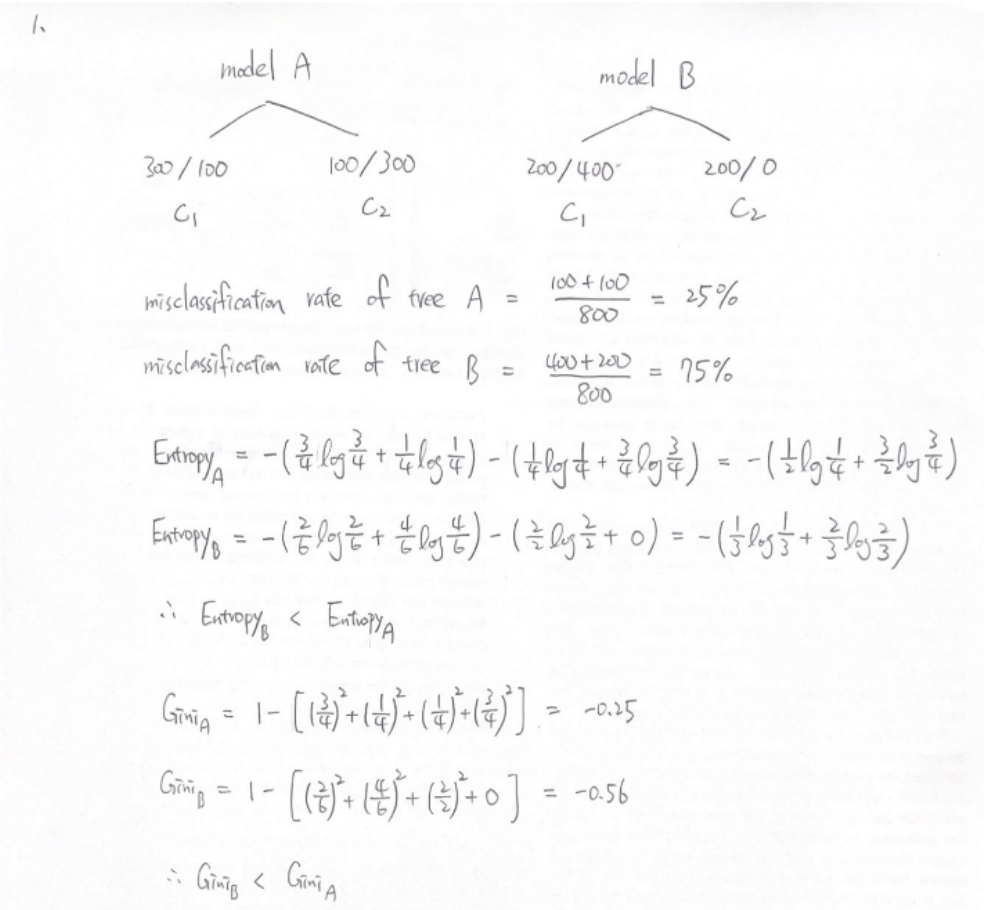
6. (30%) Tune the hyperparameter, perform feature engineering or implement more powerful ensemble methods to get a higher accuracy score. Screenshot your test score on the report. Please note that only the ensemble method can be used. The neural network method is not allowed.

```
(gradient boosting) Test-set accuarcy score: 0.8944271909999159
```

Accuracy	Your scores
acc > 0.85	30 points
0.8 < acc <= 0.85	25 points
0.7 < acc <= 0.8	20 points
acc < 0.7	0 points

Part. 2, Questions (20%):

1. (10%) Consider a data set comprising 400 data points from class C_1 and 400 data points from class C_2. Suppose that a tree model A splits these into (300, 100) at the first leaf node and (100, 300) at the second leaf node, where (n, m) denotes that n points are assigned to C_1 and m points are assigned to C_2. Similarly, suppose that a second tree model B splits them into (200, 400) and (200, 0). **Evaluate the <u>misclassification rates</u> for the two trees and hence show that they are equal**. Similarly, **evaluate the cross-entropy** $Entropy = -\sum_{k=1}^{K} p_k log_2 p_k$ and **Gini index** $Gini = 1 - \sum_{k=1}^{K} p_k^2$ **for the two trees** and show that they are both lower for tree B than for tree A. Define p_k to be the proportion of data points in region R assigned to class k, where k = 1, . . . , K



2. (10%) By making a variational minimization of the expected exponential error function given by (1) with respect to all possible functions $y(x)$, show that the minimizing function is given by (2). Define t is target variable $\in \{-1, 1\}$, x is input vector.

$$E_{x,t}[e^{-ty(x)}] = \sum_t \int e^{-ty(x)} p(t|x)p(x)\,dx \quad (1)$$

$$y(x) = \frac{1}{2} ln \frac{p(t=1|x)}{p(t=-1|x)} \quad (2)$$